



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/047,769	01/15/2002	Daniel L. Klave	SLA1062	9701

50735 7590 06/30/2006

MADSON & AUSTIN  
15 WEST SOUTH TEMPLE  
SUITE 900  
SALT LAKE CITY, UT 84101

EXAMINER
----------

FOWLKES, ANDRE R

ART UNIT	PAPER NUMBER
----------	--------------

2192

DATE MAILED: 06/30/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	Application No. 10/047,769	Applicant(s) KLAVE ET AL.	
	Examiner Andre R. Fowlkes	Art Unit 2192	

**-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

- 1) ☒ Responsive to communication(s) filed on 31 March 2006.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

- 4) ☒ Claim(s) 1, 2, 4-6, 9-20, 23-31, 33, 34 and 37-46 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1, 2, 4-6, 9-20, 23-31, 33-34 and 37-46 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)   | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### DETAILED ACTION

1. This action is in response to the amendment filed 3/31/06.
2. Claims 1, 16, and 30 have been amended. Claims 44-46 have been added.  
Claims 1, 2, 4-6, 9-20, 23-31, 33-34 and 37-46 are currently pending.

### *Claim Rejections - 35 USC § 103*

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claim 1, 2, 4-8, 11-22, 25-31, 33-36, and 39-46 rejected under 35 U.S.C. 103(a) as being unpatentable over Deo et al., (Deo), U.S. Patent No. 6,226,665, in view of Aguilar et al., U.S. Patent No. 6,446,203 (art made of record), and further in view of Parry, U.S. Patent Application Publication No. 2003/0078963 (art made of record).

As per claim 1, Deo discloses a **multi-functional peripheral configured to reduce volatile memory usage by loading individual software components**, (col. 2:46-64, "a method is defined for executing a software application on a system having a processor so as to minimize a RAM capacity required while the processor executes the software application. The method includes the step of providing application code that is

Art Unit: 2192

divided into specific software components. The software components include variables and an event handler. A first portion (comprising individual software components) of the variables and the event handler for the software application are loaded (using a loader application) from a storage memory (i.e. non-volatile memory) that is not used for execution of the application, into a RAM of the system and are executed from the RAM using the processor. Any change in a state of the system and any new event is detected by the processor while it executes the software components loaded into the RAM. In response to either a change in the state of the system or a new event, another software component is loaded into the RAM for execution by the processor, replacing at least one of the software components previously loaded. These steps repeat until execution of the software application is terminated”), **the multi-functional peripheral comprising:**

- **a processor** (col. 2:46-64, “processor”),
  - **volatile memory in electronic communication with the processor** (col. 2:46-64, “RAM (i.e. volatile memory”),
  - **non-volatile memory in electronic communication with the processor, the non-volatile memory**, (col. 2:46-64, “storage memory (i.e. non-volatile memory”),
- including:**

- **an operating system, a loader application, a plurality of individual software components** (col. 2:46-64, “a method is defined for executing a software application on a system having a processor so as to minimize a RAM capacity required while the processor executes the software application. The method includes the step of

providing application code that is divided into specific software components. The software components include variables and an event handler. A first portion (comprising individual software components) of the variables and the event handler for the software application are loaded (using a loader application) from a storage memory (i.e. non-volatile memory) that is not used for execution of the application, into a RAM of the system and are executed from the RAM using the processor. Any change in a state of the system and any new event is detected by the processor while it executes the software components loaded into the RAM. In response to either a change in the state of the system or a new event, another software component is loaded into the RAM for execution by the processor, replacing at least one of the software components previously loaded. These steps repeat until execution of the software application is terminated”, and col. 4:28-36, “Nomad includes a housing 32 and has a user interface that includes a keypad 36 disposed on the top of housing and having four buttons that are used to control the display and the functions performed by Nomad in connection with its conventional paging function and its PIM data management functions. The buttons comprise an Action button 42, a Back button 44, an Up button 38, and a Down button 40. The buttons navigate the user through a directory/menu hierarchy”, and col. 6:9-32, “Navigation on Nomad is done with hierarchical menus, enabling the user to move through a tree structure... The user moves a selection line up and down to select a desired list item (i.e. an individual software component)”),

**- instructions stored in the non-volatile memory that are executable by the processor for implementing a method comprising:**

**- loading the operating system for the multi-functional peripheral comprising a printer into the volatile memory** (col. 2:53-58, "A first portion of the variables and the event handler (i.e. operating system) for the software application are loaded (using a loader application) from a storage memory (i.e. non-volatile memory) that is not used for execution of the application, into a RAM (i.e. volatile memory) of the system"),

**- starting the operating system, loading the loader application into the volatile memory, starting the loader application, examining the loading table to determine which of the individual software components are to be loaded into the volatile memory in connection with starting the operating system** (col. 2:46-64, "a method is defined for executing a software application on a system having a processor so as to minimize a RAM capacity required while the processor executes the software application. The method includes the step of providing application code that is divided into specific software components. The software components include variables and an event handler. A first portion (comprising individual software components) of the variables and the event handler for the software application are loaded (using a loader application) from a storage memory (i.e. non-volatile memory) that is not used for execution of the application, into a RAM of the system and are executed from the RAM using the processor. Any change in a state of the system and any new event is detected by the processor while it executes the software components loaded into the RAM. In response to either a change in the state of the system or a new event, another software component is loaded into the RAM for execution by the processor, replacing at least

one of the software components previously loaded. These steps repeat until execution of the software application is terminated”),

**- individually loading each of the individual software components that are to be loaded as indicated in the loading table into the volatile memory in connection with starting the operating system** (col. 2:53-58, “A first portion of the variables and the event handler (i.e. an individual software component) for the software application are loaded (using a loader application) from a storage memory (i.e. non-volatile memory) that is not used for execution of the application, into a RAM (i.e. volatile memory) of the system”).

Deo doesn't explicitly disclose **a loading table that is directly configurable by a user to control which ... software components are loaded into volatile memory in connection with starting the operating system.**

However, Aguilar, in an analogous environment, discloses **a loading table that is directly configurable by a user to control which ... software components are loaded into volatile memory in connection with starting the operating system** (col. 2:6-11, “In the preferred embodiment, the boot code sequence responds to a specified user input sequence by presenting the user with a configuration screen suitable for altering the value of the image selection indicator such that the user may alter the boot image (i.e. the software components) that will be loaded during a subsequent execution of the boot sequence (i.e. during the starting of the operating system).”).

Therefore, it would have been obvious to a person of ordinary skill in the art, at the time the invention was made, to incorporate the teachings of Aguilar into the system of Deo to have a **loading table that is directly configurable by a user to control which ... software components are loaded into volatile memory in connection with starting the operating system.** The modification would have been obvious because one of ordinary skill in the art would have wanted a convenient way of loading different types of software at startup time (Aguilar col. 1:29-45).

Further, the Deo/Aguilar combination doesn't explicitly disclose that the multi-functional peripheral comprises **a printer**. However, Parry in an analogous environment, discloses a multi-functional peripheral device that comprises **a printer** (§ 21:14-15, "multifunctional printers, fax machines, copiers").

Therefore, it would have been obvious to a person of ordinary skill in the art, at the time the invention was made, to incorporate the teachings of Parry into the Deo/Aguilar system to incorporate a printer. The modification would have been obvious because one of ordinary skill in the art would have wanted the flexibility of loading different types of software at startup time on a printer, as well as on all other types of peripherals.

As per claim 2, the rejection of claim 1 is incorporated, and further Deo doesn't explicitly disclose that **the multi-functional peripheral is a printer/fax/copier.** However, Parry in an analogous environment, discloses a multi-functional peripheral



Art Unit: 2192

device that comprises a **printer/fax/copier** (§ 21:14-15, "multifunctional printers, fax machines, copiers").

Therefore, it would have been obvious to a person of ordinary skill in the art, at the time the invention was made, to incorporate the teachings of Parry into the Deo/Aguilar system to incorporate a printer. The modification would have been obvious because one of ordinary skill in the art would have wanted the flexibility of loading different types of software at startup time on a printer/fax/copier, as well as on all other types of peripherals.

As per claim 4, the rejection of claim 1 is incorporated, and further Deo discloses **an input component in electronic communication with the processor for a user to enter user input and thereby configure the loading table** (col. 4:28-36, "Nomad includes a housing 32 and has a user interface that includes a keypad 36 disposed on the top of housing and having four buttons that are used to control the display and the functions performed by Nomad in connection with its conventional paging function and its PIM data management functions. The buttons comprise an Action button 42, a Back button 44, an Up button 38, and a Down button 40. The buttons navigate the user through a directory/menu hierarchy").

As per claim 5, the rejection of claim 4 is incorporated, and further Deo discloses **a display in electronic communication with the processor that displays information to the user relating to the loading table** (col. 4:28-36, "Nomad includes

Art Unit: 2192

a housing 32 and has a user interface that includes a keypad 36 disposed on the top of housing and having four buttons that are used to control the display and the functions performed by Nomad in connection with its conventional paging function and its PIM data management functions. The buttons comprise an Action button 42, a Back button 44, an Up button 38, and a Down button 40. The buttons navigate the user through a directory/menu hierarchy”).

As per claim 6, the rejection of claim 5 is incorporated, and further Deo discloses **a menu structure that may be navigated by a user using the input component and the display to configure the loading table** (col. 4:28-36, “Nomad includes a housing 32 and has a user interface that includes a keypad 36 disposed on the top of housing and having four buttons that are used to control the display and the functions performed by Nomad in connection with its conventional paging function and its PIM data management functions. The buttons comprise an Action button 42, a Back button 44, an Up button 38, and a Down button 40. The buttons navigate the user through a directory/menu hierarchy”).

As per claim 11, the rejection of claim 1 is incorporated, and further Deo discloses that **the volatile memory is RAM** (col. 2:48, “RAM”).

As per claim 12, the rejection of claim 1 is incorporated, and further Deo discloses that **the individual software components are software libraries** (col. 2:51, "software components (i.e. software libraries)").

As per claim 13, the rejection of claim 1 is incorporated, and further Deo discloses **a communications module in electronic communication with the processor for communications with a computer** (col. 13:46-48, "Also coupled to the object maker is a network interface 266, which provides communication to other computers (via the web)"), **and a web interface accessible by a user through use of a web browser to configure the loading table** (col. 13:46-48, "Also coupled to the object maker is a network interface 266, which provides communication to other computers (via the web)").

As per claim 14, the rejection of claim 13 is incorporated, and further Deo discloses that **the web interface comprises a web page** (col. 13:46-48, "Also coupled to the object maker is a network interface 266, which provides communication to other computers (via the web)").

As per claim 15, the rejection of claim 1 is incorporated, and further Deo discloses **examining hardware configuration by the loader application and modifying the loading table based on the hardware configuration** (col. 13:55-65, "By executing an applet, the basic functionality of Nomad (or any other small portable

Art Unit: 2192

device in which the present invention is embodied) can be substantially altered and thus expanded. Accordingly, the useful life of the device is extended by providing appropriate applets to enhance its functional capabilities. For example, if embodied in a cell phone that when originally sold, did not include the ability to provide caller identification (ID), an appropriate applet executed by a processor in the cell phone could provide that additional caller ID functionality”).

As per claims 16-22 and 25-29, this is a computer readable medium version of the claimed system discussed above, in claims 1, 2, 4, 6-8 and 11-15, wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see the Deo/Aguilar/Parry system (Deo col. 2:45-3:50, Aguilar col. 2:6-11 and Parry col. 21:14-15).

As per claims 30, 31, 33-36 and 39-43, this is a method version of the claimed system discussed above, in claims 1, 2, 4, 6-8 and 11-15, wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see the Deo/Aguilar/Parry system (Deo col. 2:45-3:50, Aguilar col. 2:6-11 and Parry col. 21:14-15).

As per claim 44, the rejection of claim 1 is incorporated, and further Deo discloses that **the individual software components are loaded into volatile memory after starting the operating system**, (col. 3:18-23, “An operating system kernel ...

Art Unit: 2192

ensures that the required software components are loaded when needed", and the operating system kernel is functional only after starting the operating system).

As per claim 45, this is a computer readable medium version of the claimed system discussed above, in claim 44, wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see the Deo/Aguilar/Parry system (Deo col. 2:45-3:50, Aguilar col. 2:6-11 and Parry col. 21:14-15).

As per claim 46, this is a computer readable medium version of the claimed system discussed above, in claim 44, wherein all claimed limitations have also been addressed and/or cited as set forth above. For example, see the Deo/Aguilar/Parry system (Deo col. 2:45-3:50, Aguilar col. 2:6-11 and Parry col. 21:14-15).

4. Claims 9, 10, 23, 24, 37 and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Deo et al., (Deo), U.S. Patent No. 6,226,665, in view of Aguilar et al., U.S. Patent No. 6,446,203 (art made of record), further in view of Parry, U.S. Patent Application Publication No. 2003/0078963 (art made of record) and further in view of Buxton et al., (Buxton), U.S. Patent No. 5,970,252.

As per claim 9, the rejection of claim 1 is incorporated, and further the Deo/Aguilar/Parry system doesn't explicitly disclose that **the loading table is a license table comprising a list of licenses relating to the individual software components.**

However, Buxton, in an analogous environment, discloses that **the loading table is a license table comprising a list of licenses relating to the individual software components** (col. 18:1-3, "If a component's certification is not on the list, it is assumed that the component is unlicensed and therefore is not loaded and its use is unauthorized").

Therefore, it would have been obvious to a person of ordinary skill in the art, at the time the invention was made, to incorporate the teachings of Buxton into the Deo/Aguilar/Parry system to have **the loading table is a license table comprising a list of licenses relating to the individual software components**. The modification would have been obvious because one of ordinary skill in the art would have wanted to ensure that the component is properly licensed and is used within the terms and conditions (Buxton 17:18-47).

As per claim 10, the rejection of claim 9 is incorporated, and further the Deo/Aguilar/Parry system doesn't explicitly disclose that **the individual software components with licenses, as indicated by the license table, are loaded into the volatile memory**.

However, Buxton, in an analogous environment, discloses that **the individual software components with licenses, as indicated by the license table, are loaded into the volatile memory** (col. 18:1-3, "If a component's certification is not on the list, it is assumed that the component is unlicensed and therefore is not loaded and its use is

Art Unit: 2192

unauthorized, (software components with licenses are loaded into the volatile memory)").

Therefore, it would have been obvious to a person of ordinary skill in the art, at the time the invention was made, to incorporate the teachings of Buxton into the Deo/Aguilar/Parry system to have **the individual software components with licenses, as indicated by the license table, are loaded into the volatile memory.** The modification would have been obvious because one of ordinary skill in the art would have wanted to ensure that the component is properly licensed and is used within the terms and conditions (Buxton 17:18-47).

As per claims 23, 24, 37 and 38, the Deo/Aguilar/Parry/Buxton combination also discloses such claimed limitations as addressed in claims 9-10 above, respectively. For example, see the Deo/Aguilar/Parry/Buxton system (Deo col. 2:45-3:50, Aguilar col. 2:6-11, Parry col. 21:14-15 and Buxton 17:18-47).

### ***Response to Arguments***

5. Applicants arguments have been considered but they are not persuasive.

*In the remarks, the applicant has argued substantially that:*

1) The cited art does not disclose individually loading each of the individual software components, at p. 11:12-12:5, 13:18-20, 14:9-23 and 15:17-19.

Art Unit: 2192

*Examiner's response:*

1) The examiner disagrees with applicant's characterization of the applied art. Deo discloses individually loading each of the individual software components, at col. 2:53-58, "A first portion of the variables and the event handler (i.e. an individual software component) for the software application are loaded (using a loader application) from a storage memory (i.e. non-volatile memory) that is not used for execution of the application, into a RAM (i.e. volatile memory) of the system," as addressed in the above art rejection.

*In the remarks, the applicant has argued substantially that:*

2) The cited art does not disclose a loading table that is directly configurable by the user to determine which of the individual software components are loaded into the volatile memory. Applicant specifically argues that Aguilar does not disclose determining which individual software components are loaded into volatile memory, at 12:11-14, 13:1-20 and 16:4-5.

*Examiner's response:*

2) In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). In this case, Aguilar is cited to disclose using a loading table to determine which



software components are loaded (Aguilar col. 2:6-11). Deo is cited to disclose individually loading software components, as addressed above in the examiner's response to applicant's first argument.

*In the remarks, the applicant has argued substantially that:*

3) The cited art does not disclose the limitation of new claims 44-46: that the individual software components are loaded into volatile memory after starting the operating system, at p. 15:11-15.

*Examiner's response:*

3) The examiner disagrees with applicant's interpretation of the applied art. Deo does disclose that the individual software components are loaded into volatile memory after starting the operating system, at col. 3:18-23, "An operating system kernel ... ensures that the required software components are loaded when needed", and the operating system kernel is functional only after starting the operating system.

### **Conclusion**

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Andre R. Fowlkes whose telephone number is (571) 272-3697. The examiner can normally be reached on Monday - Friday, 8:00am-4:30pm.

Art Unit: 2192

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571)272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

ARF



TUAN DAM  
SUPERVISORY PATENT EXAMINER